

AMENDMENTS TO THE CLAIMS

Please amend Claims 1 and 16 as follows:

- 1 1. (Currently Amended) A process for routing packets through a load balancing array
2 of servers across a network in a computer environment, comprising the steps of:
3 providing a plurality of load balancing servers;
4 providing at least one back end Web server;
5 wherein one of said load balancing servers is also a scheduler;
6 wherein all request packets from [[a]] clients destined for the load balancing array
7 are [[is]] routed through said scheduler;
8 wherein said scheduler routes and load balances ~~said~~ a request packet to a load
9 balancing server;
10 wherein said load balancing server routes and load balances said request packet to a
11 back end Web server;
12 wherein said back end Web server's response packet to said request packet is sent to
13 said load balancing server; and
14 wherein said load balancing server sends said response packet directly to said client.

- 1 2. The process of Claim 1, wherein said scheduler routes and load balances client
2 requests to itself.

- 1 3. The process of Claim 1, further comprising the steps of:
2 detecting the failure of said scheduler; and
3 electing one of said load balancing servers as the new scheduler.

1 4. The process of Claim 1, wherein said scheduler detects the failure of other load
2 balancing servers; and wherein said scheduler stops routing packets to any failed load
3 balancing servers.

1 5. The process of Claim 1, wherein said load balancing server schedules sessions to
2 back end Web servers based on a cookie or session ID.

1 6. The process of Claim 1, wherein said load balancing server uses cookie injection to
2 map a client to a specific back end Web server.

1 7. The process of Claim 1, wherein said load balancing server decrypts said request
2 packet if it is an SSL session before routing and load balancing said request packet to a back
3 end Web server.

1 8. The process of Claim 7, wherein said load balancing server encrypts said response
2 packet if it is an SSL session before sending said response packet directly to said client.

1 9. The process of Claim 1, wherein said load balancing server establishes a connection
2 with said client and said client keeps said connection alive with said load balancing server.

1 10. The process of Claim 9, wherein said load balancing server performs URL based
2 scheduling of request packets.

1 11. The process of Claim 9, wherein said load balancing server performs hash
2 scheduling of request packets.

1 12. The process of Claim 1, wherein said load balancing server maintains persistent
2 connections in all its paths when required; and wherein said load balancing server uses hash
3 group based persistence to maintain its persistence tables.

1 13. The process of Claim 1, wherein said load balancing server detects if a back end
2 Web server fails; and wherein said load balancing server stops routing request packets to
3 failed back end Web servers.

1 14. The process of Claim 1, further comprising the step of:
2 providing a content delivery network; and
3 wherein said load balancing server modifies select URLs in the HTML page in said
4 response packet to serve them from said content delivery network.

1 15. The process of Claim 14, wherein HTML pages that have modified URLs are cached
2 to improve performance.

1 16. (Currently Amended) An apparatus for routing packets through a load balancing
2 array of servers across a network in a computer environment, comprising:
3 a plurality of load balancing servers;
4 at least one back end Web server;
5 wherein one of said load balancing servers is also a scheduler;

6 wherein all request packets from [[a]] clients destined for the load balancing array .
7 are [[is]] routed through said scheduler;

8 wherein said scheduler routes and load balances ~~said~~ a request packet to a load
9 balancing server;

10 wherein said load balancing server routes and load balances said request packet to a
11 back end Web server;

12 wherein said back end Web server's response packet to said request packet is sent to
13 said load balancing server; and

14 wherein said load balancing server sends said response packet directly to said client.

1 17. The apparatus of Claim 16, wherein said scheduler routes and load balances client
2 requests to itself.

1 18. The apparatus of Claim 16, further comprising:
2 a module for detecting the failure of said scheduler; and
3 a module for electing one of said load balancing servers as the new scheduler.

1 19. The apparatus of Claim 16, wherein said scheduler detects the failure of other load
2 balancing servers; and wherein said scheduler stops routing packets to any failed load
3 balancing servers.

1 20. The apparatus of Claim 16, wherein said load balancing server schedules sessions to
2 back end Web servers based on a cookie or session ID.

1 21. The apparatus of Claim 16, wherein said load balancing server uses cookie injection
2 to map a client to a specific back end Web server.

1 22. The apparatus of Claim 16, wherein said load balancing server decrypts said request
2 packet if it is an SSL session before routing and load balancing said request packet to a back
3 end Web server.

1 23. The apparatus of Claim 22, wherein said load balancing server encrypts said
2 response packet if it is an SSL session before sending said response packet directly to said
3 client.

1 24. The apparatus of Claim 16, wherein said load balancing server establishes a
2 connection with said client and said client keeps said connection alive with said load
3 balancing server.

1 25. The apparatus of Claim 24, wherein said load balancing server performs URL based
2 scheduling of request packets.

1 26. The apparatus of Claim 24, wherein said load balancing server performs hash
2 scheduling of request packets.

1 27. The apparatus of Claim 16, wherein said load balancing server maintains persistent
2 connections in all its paths when required; and wherein said load balancing server uses hash
3 group based persistence to maintain its persistence tables.

1 28. The apparatus of Claim 16, wherein said load balancing server detects if a back end
2 Web server fails; and wherein said load balancing server stops routing request packets to
3 failed back end Web servers.

1 29. The apparatus of Claim 16, further comprising:
2 a content delivery network; and
3 wherein said load balancing server modifies select URLs in the HTML page in said
4 response packet to serve them from said content delivery network.

1 30. The apparatus of Claim 29, wherein HTML pages that have modified URLs are
2 cached to improve performance.